GUIA RAPIDA DEL BASIC MSX 1 y 2



Claves para la utilización de la GUIA.

* Todos los textos escritos en mayúsculas deben ser transcritos literalmente.

* Los textos que se encuentran entre llaves [ ] indican características opcionales y que pueden por tanto, ser omitidas.

* Los textos separados por una barra vertical (|) son optativos. Se tiene que optar por uno sólo de los textos que se incluyen.

* Precede un arterisco (*) a todas las instrucciones existentes únicamente en los MSX de segunda generación.

* Precede una **D** a todas las instrucciones existentes únicamente en los MSX con unidad de disco, sean de la primera o de la segunda generación.

* Todos los demás símbolos (comillas, paréntesis, corchetes, etc) debe transcribirse literalmente al utilizar las instrucciones.

* Existen ciertas palabras clave que a continuación comentamos.

<num> indica cualquier expresión numérica, es decir, números, variables numéricas, y resultados de expresiones complejas.

<cadena> indica cualquier expresión de tipo alfanumérico, textos entrecomillados, variables de cadena, y resultados de operaciones con cadenas.

.ext indica una extensión al nombre de un fichero. Se utiliza en los nombres de fichero creados sobre discos para distinguirlos por grupos. Por ejemplo .BAS para todos los ficheros que contengan programas en BASIC.

car indica que se debe incluir un único carácter. Si se incluye una cadena se tomará únicamente el primer carácter.

const indica un valor constante, sea numérico o alfanumérico. No se admiten por tanto variables ni expresiones o resultados de operaciones.

dirección o **dir** indica una dirección de memoria válida, entre Ø y 65535. Para utilizar correctamente las intrucciones que utilizan direcciones es conveniente conocer el mapa de memoria de los MSX.

dispositivo indica uno de los dispositivos de grabación de datos que haya conectados al ordenador. Son válidos A:, B:, C:,... para las unidades de disco, GRP: para la pantalla de gráficos, CRT: para la pantalla de textos, AUX: para la puerta serie, LPT: para la impresora paralela, CAS: para la cinta de cassette y MEM: para el RAM-DISK de los MSX-2.

incremento indica una cantidad que se añadirá progresivamente a otra (usada por ejemplo en AUTO o FOR... NEXT).

numlin (con o sin llaves) indica número de línea. Sólo puede sustituirse por un número de línea válido. No se aceptan variables ni expresiones numéricas

numfich (con o sin llaves) indica un número de fichero, ya que dentro del programa (y tras haber hecho un OPEN del fichero) nos referimos a cada fichero únicamente por su número.

Ejemplo:

Si encomtramos:

BLOAD"[dispositivo:][nombre[.ext]]"[,desplazamiento][,R]

esto significa que debemos escribir siempre BLOAD y comillas. El nombre de dispositivo es opcional. El nombre es también opcional (puede ser omitido), y sólo si se ha escrito el nombre podemos escribir la extensión (opcionalmente también).

Tanto el desplazamiento como la opción R son opcionales, y ¡atención a los detalles!, si no indicamos desplazamiento no hay que incluir la coma (ya que se halla dentro de las llaves).

LISTA DE INSTRUCCIONES

ABS (<num>)
Retorna el valor absoluto del dato especificado.

ASC (<cadena>)
Retorna el código ASCII del primer carácter de la cadena.

ATN (<num>)
Retorna el valor del arcotangente del valor indicado.

ASC numlin, incremento
Genera automáticamente los números de línea de programa a partir de la línea especificada y con el incremento indicado.

BASE (<num>)
Lee o escribe un dato en la tabla de direcciones del VDP. (Cambia la dirección de una de las tablas de pantalla).

BEEP Genera un pitido de aviso.

BIN$ (<num>)
Retorna en formato alfanumérico la representación binaria del valor especificado.

Ⓓ **BLOAD** [dispositivo:] [nombre[.ext]]"[,R][,desplazamiento]
Lee del dispositivo indicado un fichero en formato binario. Opcionalmente lo autoejecuta, o lo lee desplazado el número de posiciones de memoria indicado (positivo o negativo).

Ⓓ **BLOAD** "[unidad:]nombre[.ext]",S[,desplazamiento]
Lee del disco indicado un fichero en formato binario y lo ubica en la vídeo-RAM. Opcionalmente lo lee desplazado el número de posiciones de memoria indicado (positivo o negativo).

CALL instrucción
Ejecuta una instrucción extendida residente en un cartucho ROM.

CDBL (<num>)
Convierte el dato indicado (de cualquier tipo numérico) a doble precisión.

CHR$ (<num>)
Retorna el carácter correspondiente al código especificado.

CINT (<num>)
Convierte el dato indicado (de cualquier tipo numérico) a formato entero.

CIRCLE (coor.X,coor,Y),radio,[color],[ángulo],[ángulo],[Y/X]
Dibuja un círculo, elipse, o arco con centro en el punto indicado, con el radio y color especificados. Si se indican los ángulos de inicio y final se realiza un arco, y si éstos son negativos un abanico. La relación Y/X (Aspect Ratio) permite modificar el grado de achatamiento de la figura. (circunferencia=4/3).

CLEAR <num1>,<num2>
Inicializa todas las variables y asigna para la zona de cadenas una memoria de num1 bytes.
Indica al BASIC que la dirección de memoria más alta a la que puede acceder es num2.
Cierra todos los ficheros abiertos, si hay alguno.

CLOAD ["nombre"]
Carga de la cinta el programa BASIC con el nombre indicado.

CLOAD? ["nombre"]
Verifica que el programa indicado de la cinta de cassette sea igual al almacenado en memoria.

CLOSE [#][<num1>[,<num2>]...
Cierra los ficheros con los números num1, num2,...

CLS Borra la pantalla.

COLOR frente,fondo, borde
Selecciona los colores de texto/gráficos, fondo y borde según se indique.

⊙ **COLOR** =(paleta, rojo, verde, azul)
Asigna al color de la paleta especificado un nuevo color según la intensidad de rojo, verde y azul especificada.

⊙ **COLOR[=NEW]**
Devuelve la paleta de colores a sus valores iniciales.

⊙ **COLOR=RESTORE**
Traslada a los registros del VDP la paleta de colores existente en la VRAM.

⊙ **COLOR SPRITE** (plano)=paleta
Asigna el color del número de paleta especificado al plano de sprites indicado.

⊙ **COLOR SPRITE$** (plano)=<cadena>
Especifica los colores correspondientes al sprite, uno para cada línea horizontal del mismo.

CONT
Continúa la ejecución de un programa tras haber sido detenido con STOP (o con CTRL+STOP).

Ⓓ **COPY** "Unidad1:]nombre1"TO"[unidad2:]nombre2"
Copia el fichero nombre1 de la unidad1 sobre la unidad2 con el nombre de fichero especificado.

⊙ **COPY** (x1,y1)−(x2,y2)[,página1] TO (x3,y3),[página2],[op.lógica]
Copia el gráfico incluido en el recuadro (x1,y1)−(x2,y2) de la página1 indicada a partir del punto (x3,y3) de la página 2 especificada, realizando opcionalmente la operación lógica indicada.

⊙ **COPY** (x1,1y)−(x2,y2)[,página] TO "[unidad:]nombre"
Copia el gráfico incluido en el recuadro (x1,y1)−(x2,y2) de la página especificada en el fichero indicado de la unidad especificada.

⊙ **COPY** (x1,y1)−(x2,y2)[página] TO variable-matriz
Copia el gráfico incluido en el recuadro (x1,y1)−(x2,y2)de la página especificada en una variable de matriz.

⊙ **COPY** "[unidad:]nombre"[,dirección] TO (x,y)[,página],[op. lógica]
Copia el fichero especificado en la posición (x,y) de la página indicada. Opcionalmente se realiza una operación lógica.

⊙ **COPY** variable-matriz TO (x,y),[página],[op. lógica]
Copia el gráfico almacenado en la variable de matriz en la posición (x,y) de la página indicada. Opcionalmente se realiza una operación lógica.

⊙ **COPY** "[unidad:]nombre" TO variable-matriz
Copia el gráfico almacenado en el fichero indicado a la variable de matriz.

⊙ **COPY** variable-matriz TO "[unidad:]nombre"
Copia el gráfico almacenado en la variable de matriz al fichero indicado.

⊙ **COPY SCREEN** [modo],[máscara]
Digitaliza una señal de vídeo externa y la almacena en la RAM de vídeo.

COS (<num>)
Retorna el coseno del valor especificado, que debe estar en radianes.

CSAVE "nombre"[,velocidad]
Almacena en una cinta de cassette el programa actualmente en memoria. Opcionalmente se puede variar la velocidad de grabación entre 1200 (1) y 2400 (2) baudios.

CSNG <num>)
Convierte el dato numérico especificado a simple precisión.

CSRLIN
Retorna el número de línea en que se encuentra el cursor de texto.

Ⓓ **CVD** (<cadena>)
Convierte la cadena de caracteres en un dato numérico de doble precisión. (La cadena debe estar compactada con MKD$).

Ⓓ **CVI** (<cadena>)
Convierte la cadena de caracteres en un dato numérico entero. (La cadena debe estar compactada con MKI$).

Ⓓ **CVS** (<cadena>)
Convierte la cadena de caracteres en un dato numérico de simple precisión. (La cadena debe estar compactada con MKS$).

DATA const[,const]...
Ofrece una serie de datos constantes que serán leídos mediante una sentencia READ.

DEF FN nombre[(parámetro1[,parámetro2]...)] =expresión
Define una función realizada por el usuario.

DEBL car[−car][,car[−car]]...
Toma todas las variables que comienzan con los caracteres indicados como variables de doble precisión.

DEFINT car[−car][,car[−car]]...
Toma todas las variables que comienzan con los caracteres indicados como variables enteras.

DEFSNG car[−car][,car[−car]]...
Toma todas las variables que comienzan con los caracteres indicados como variables de simple precisión.

DEFR car[−car][,car[−car]]...
Toma todas las variables que comienzan con los caracteres indicados como variables de tipo cadena (string).

DEFUSR[x]=dirección
Define la dirección de ejecución de una rutina en lenguaje máquina que se ejecutará con la instrucción USR. x es opcional y puede variar entre Ø y 9.

DELETE [numlin][−numlin]
Borra la línea o grupo de líneas especificadas del programa residente actualmente en memoria.

DIM matriz1 (limite1[,limite2]...), matriz2(...)...
Define el nombre, dimensión, tamaño y tipo de una matriz, y reserva espacio para ella en la memoria.

DRAW lista de mandatos
Ejecuta la serie de mandatos gráficos del GML incluidos en la lista.

Ⓓ **DSKF**(unidad)
Retorna el espacio libre utilizable que queda en el disco, expresado en clusters.

END
Da por terminada la ejecución del programa en curso. Cierra todos los ficheros abiertos si los hay.

Ⓓ **EOF**(numfich)
Retorna un valor lógico TRUE (−1) si se ha alcanzado el final del archivo indicado. En otro caso retorna FALSE (Ø).

ERASE var[,var]...
Borra de la memoria la o las variables de matriz indicadas. Tras esta instrucción pueden volver a definirse con DIM.

ERL
Retorna el número de línea en que se produjo el último error.

ERR
Retorna el código del último error que se haya producido.

ERROR <num>
Simula el error que tiene por código el especificado.

EXP(<num>)
Retorna el valor del exponencial del número indicado, es decir, el número "e" elevado al valor dado.

Ⓓ **FIELD** [#] numfich, longi AS <cadena>[,long2 AS <cadena>]...
Especifica el formato de cada una de las fichas (registros) que componen un fichero de acceso directo.

Ⓓ **FILES** ["[unidad:][nombre[.ext]]"]
Lista en pantalla todos los archivos del disco que concuerden con la especificación indicada. En caso de no dar ninguna especificación se listarán todos los archivos del disco.

FIX(<num>)
Retorna el valor truncado del número indicado (su valor sin decimales).

FOR var = <num1> TO <num2> [STEP incremento]
Repite la ejecución del conjunto de instrucciones situadas entre el FOR y su correspondiente NEXT.

FRE(<num>)
Retorna la cantidad de memoria libre en la memoria MSX utilizable para la confección y ejecución de programas en BASIC.

FRE("")
Retorna la cantidad de memoria libre para la utilización de cadenas de caracteres.

Ⓓ **GET** [#]numfich[,numreg]
Lee del fichero indicado (sólo en disco) el registro especificado y lo almacena según se indicase en la definición del FIELD. Si no se incluye número de registro se leerá el primero del fichero, o el siguiente al anteriormente leído.

⊙ **GET DATE** <cadena>[,A]
Introduce en la cadena especificada la fecha actual indicada por el chip de reloj. La opción A permite emplear la alarma.

⊙ **GET TIME** <cadena>[,A]
Introduce en la cadena especificada la hora actual, indicada por el chip de reloj. La opción A permite emplear la alarma.

GOSUB numlin
Transfiere el control del programa a la línea especificada y memoriza la línea actual como dirección de retorno de subrutina.

GOTO numlin
Transfiere el control del programa a la línea especificada.

HEX$(<num>)
Retorna en formato cadena el valor hexadecimal del número indicado.

IF condición **THEN** instrucción [**ELSE** instrucción]
De acuerdo con la veracidad o falsedad de la condición se ejecuta la instrucción que sigue al THEN (se cumple la condición) o la que sigue al ELSE (no se cumple la condición).

INKEY$
Retorna el carácter correspondiente a la tecla que se esté pulsando y la cadena nula ("") si no se está pulsando ninguna tecla.

INP(puerto)
Lee un byte del puerto de entrada salida indicado.

INPUT ["mensaje";]var1[,var2]...
Insta al usuario a introducir valores desde el teclado y los asigna a cada una de las variables indicadas.

INPUT$ (<num>)
Espera a que se tecleen tantos caracteres como se hayan indicado y se retorna la cadena que forman éstos.

INPUT$ (<num>,[#]numfich)
Lee tantos caracteres como se indiquen del fichero especificados y los retorna en formato cadena.

INPUT#numfich, var1[,var2]...
Lee un dato del fichero indicado y lo asigna a la variable correspondiente.

INSTR([<num>,]<cadena1>,<cadena2>)
Busca la cadena1 dentro de la cadena2. En caso de localizarla, retorna el primer lugar en que esto ocurre. En caso contrario retorna Ø. Si se especifica el número opcional, la búsqueda comienza a partir de la posición indicada.

INT(<num>)
Retorna la parte entera del número indicado.

INTERVAL ON|OFF|STOP
Válida, inválida o retiene las interrupciones generadas mediante ON INTERVAL GOSUB.

KEY numtec,<cadena>
Asigna a la tecla de función cuyo número se indica, una cadena de caracteres.

KEY LIST
Lista el contenido de todas las teclas de función.

KEY ON|OFF
Hace aparecer o desaparecer las teclas de función de la parte inferior de la pantalla.

KEY (<num>) ON|OF|STOP
Habilita, inhabilita o retiene una interrupción generada por la pulsación de una tecla de función.

Ⓓ **KILL** "[unidad:]nombre3[.ext]"
Borra el fichero indicado del disco.

LEFT$(<cadena>,<num>)
Retorna en formato cadena los <num> caracteres situados más a la izquierda de la cadena indicada.

LEN(<cadena>)
Retorna la longitud (número de caracteres) de la cadena.

LET <var>=valor
Asigna a la variable indicada el valor que se expresa a la derecha del signo igual. Esta instrucción puede omitirse si se desea.

LINE [(x1,y1)]−(x2,y2),[color],[B|BF],[op. lógica]
Dibuja una línea o un rectángulo entre los puntos indicados. En los MSX-2 puede indicarse opcionalmente una operación lógica a realizarse.

LINE INPUT ["comentario";]var
Espera la entrada desde el teclado de la variable de cadena. Se toma como entrada todo lo escrito hasta que se pulse RETURN (hasta 254 caracteres).

LINE INPUT # numfich, var
Lee del fichero indicado una cadena de caracteres (hasta que se halle un RETURN) de hasta 254 caracteres.

LIST [numlin][−numlin]
Visualiza por pantalla el listado del programa entre las líneas indicadas.

LLIST [numlin][−numlin]
Imprime por impresora el listado del programa entre las líneas indicadas.

LOAD "[dispositivo:][nombre[.ext]]"[,R]
Carga en memoria un programa en BASIC del dispositivo indicado. Si el dispositivo es la cinta de cassette, dicho programa debe estar en formato ASCII.

⊙ **LOC**(numfich)
Indica la posición actual del puntero de lectura en un determinado fichero. Si éste es secuencial ofrece el registro (grupo de 256 caracteres) actual, en caso de ser de acceso aleatorio ofrece el número del último registro leído.

LOCATE [coor.X],[coor.Y],[interruptor]
Coloca el cursor de texto en la posición de la pantalla indicada. Si se especifica el interruptor (Ø ó 1) desaparece o aparece el cursor respectivamente.

LOF(numfich)
Ofrece la longitud en bytes de fichero indicado.

LOG(<num>)
Retorna el valor del logaritmo natural del valor indicado.

LPOS(<num>)
Retorna la posición de la cabeza de impresión de la impresora. El valor numérico es indicado es indiferente.

LPRINT [<exp>][,|;][<exp>]...
Escribe por impresora las expresiones con el formato indicado.

LPRINT USING formato;<exp>[,|;][<exp>]...
Escribe por impresora las expresiones con el formato indicado.

Ⓓ **LSET** <cadena1>=<cadena2>
Escribe los datos de cadena2 en el registro indicado por cadena1 de un fichero aleatorio, justificándolo a la izquierda.

MAXFILES=<num>
Declara el número máximo de ficheros que pueden permanecer abiertos simultáneamente.

MERGE "[dispositivo:][nombre[.ext]]"
Lee del dispositivo indicado un programa en formato ASCII y lo fusiona con el ya existente en la memoria del ordenador. En caso de haber líneas en común se utilizan las del programa proveniente del dispositivo.

MID$(<cadena>,inicio[,num])
Retorna los num primeros caracteres de la cadena indicada a partir de la posición de inicio.

MID$(<cadena1>,inicio[,num])=<cadena2>
Sustituye los num primeros caracteres de la cadena1 a partir de la posición de inicio por la cadena2.

Ⓓ **MKD$**(<num>)
Compacta el número de doble precisión en una cadena.

Ⓓ **MKI$**(<num>)
Compacta el número entero en una cadena.

Ⓓ **MKS$**(<num>)
Compacta el número de simple precisión en una cadena.

MOTOR [ON|OFF]
Pone en marcha o desconecta el motor del aparato de cassette.

Ⓓ **NAME** "[unidad:]nombre[.ext]" AS "[unidad:]nombre[.ext]"
Cambia el nombre de un fichero determinado en la unidad de disco.

NEW
Borra un programa en BASIC de la memoria del ordenador y anula todas las variables.

NEXT [variable]
Cierra un bucle FOR... NEXT. La variable debe ser la misma que ha sido indicado anteriormente en la línea FOR.

OCT$(<num>)
Retorna en base octal (base 8), en formato cadena, el valor indicado.

ON ERROR GOTO <numlin>
Desvía la ejecución del programa a la línea indicada en caso de que se produzca cualquier error.

ON <num> **GOSUB** <numlin>[,<numlin>]...
Desvía la ejecución del programa a una de las líneas indicadas según sea el valor de la expresión numérica inicial. El salto es a una subrutina, es decir, se puede retornar a este punto con una instrucción RETURN.

ON <num> **GOTO**<numlin>[,<numlin>]...
Desvía la ejecución del programa a una de las líneas indicadas según sea el valor de la expresión numérica inicial.

ON INTERVAL=tiempo **GOSUB** <numlin>
Indica al intérprete BASIC que debe desviar la ejecución del programa a la subrutina indicada cada vez que pase el intervalo de tiempo indicado (medido en 1/5Ø de segundo).

ON KEY GOSUB <numlin>[,<numlin>]...
Indica al intérprete BASIC que debe desviar la ejecución del programa a una de las subrutinas indicadas cada vez que se pulse una tecla de función.

ON SPRITE GOSUB <numlin>
Indica al intérprete BASIC que debe desviar la ejecución del programa a la subrutina cada vez que se produzca una colisión entre sprites.

ON STOP GOSUB <numlin>
Indica al intérprete BASIC que debe desviar la ejecución del programa a la subrutina indicada cada vez que se pulse CTRL+STOP.

ON STRIG GOSUB <numlin>[,<numlin>]...
Indica al intérprete BASIC que debe desviar la ejecución del programa a la subrutina indicada cada vez que se pulse la barra del espacio o el botón de disparo de un joystick, ratón o bola gráfica.

OPEN "[dispositivo][nombre[.ext]]" [FOR modo] AS [#] numfich [LEN=longreg]
Abre un fichero en el modo indicado, le asigna un número de fichero y, opcionalmente define la longitud del registro (sólo ficheros aleatorios en disco).

OUT puerto,<num>
Envía por el puerto indicado un valor numérico (un byte).

PAD(<num>)
Retorna el estado de un cierto periférico conectado en la salida de joystick (paddle, lápiz óptico, ratón o bola gráfica).

PAINT (coor.X,coor.Y),[pintura],[contorno]
Colorea el área que contiene el punto indicado con el color de pintura. En modos de pantalla superiores al 3, puede especificarse el color del que debe ser el contorno de la figura a rellenar. En caso de omitirse se toma el color de pinutra como color de contorno.

PDL(<num>)
Retorna el valor indicativo de la posición del paddle conectado a uno de los puertos de joystick.

PEEK(<dir>)
Retorna el byte contenido en la posición de memoria cuya dirección se indica.

PLAY [comando1],[comando2],[comando3]
Envía el sonido generado por cada uno de los comandos a los respectivos canales de sonido: 1, 2 ó 3.

PLAY(canal)
Retorna un valor lógico TRUE (−1) si se está ejecutando algún sonido por el canal indicado y FALSE (Ø) en caso contrario. Un Ø en lugar del canal realiza la comprobación en todos los canales.

POINT(coor.X,coor.Y)
Retorna el código del color con que aparece en la pantalla el punto indicado.

POKE dir,<num>
Introduce en el byte indicado por la dirección de memoria un valor numérico.

POS(<num>)
Retorna la posición horizontal del cursor en la pantalla. El dato numérico es ignorado.

PRESET(coor.X,coor.Y),[color],[op. lógica]
Dibuja o borra de la pantalla el punto indicado. Opcionalmente se puede indicar una operación lógica (sólo MSX-2).

PRINT [expresión][,;][expresión]...
muestra el pantalla las expresiones indicadas (sólo válido en modo texto).

PRINT USING formato;expresión[,expresión]...
Muestra en pantalla las expresiones indicadas según el formato especificado.

PRINT #numfich,[expresión][,/;][expresión]...
Envía las expresiones al fichero indicado.

PRINT #numfich **USING** formato;expresión[,expresión]...
Envía las expresiones al fichero con el formato especificado.

PSET (coor.X,coor.Y),[color],[op.lógica]
Dibuja un punto en la pantalla con el color indicado. Opcionalmente se puede realizar una operación lógica (sólo MSX-2).

Ⓓ **PUT** [#]numfich[,numreg]
Transcribe al disco el registro actualmente en memoria en la posición numreg del fichero. Si no se indica ninguna posición se coloca a continuación del último registro grabado o leído.

PUT SPRITE plano,(coor.X,coor.Y),[color],[n.° sprite]
Coloca en la pantalla, en la posición especificada, el sprite indicado en un plano determinado.

READ <var>[,<var>]...
Lee datos de una sentencia DATA y los asigna a las variables indicadas.

REM comentario
Sentencia de comentario en un programa. El intérprete BASIC ignorará todo lo que hay detrás de la palabra REM.

RENUM [nuevo inicio],[antiguo inicio],[incremento]
Renumera las líneas del programa en memoria. Cambia el número de la línea antiguo inicio a nuevo inicio, y a partir de este punto, numera cada línea sumándole el incremento al número de la línea anterior.

RESTORE [<numlin>]
Indica en qué línea deben comenzar a leerse las sentencias DATA.

RESUME [Ø|NEXT|<numlin>]
Retorna al programa principal tras ejecutar una rutina de proceso de errores. Si se indica Ø (o no se indica nada) la ejecución continúa en el punto en que se produjo el error. Si se indica NEXT, en la línea siguiente, y si se indica un número de línea, la ejecución continúa en dicho lugar.

RETURN [<numlin>]
Retorna al programa principal desde una subrutina. Si no se especifica nada, el control continúa en la instrucción que siguiera al GOSUB que saltó a la subrutina. Si se indica un número de línea, la ejecución continúa en ese punto.

RIGHT$(<cadena>,<num>)
Retorna los num caracteres más a la derecha de la cadena indicada.

RND(<num>)
Retorna un número aleatorio positivo entre Ø y 1.

Ⓓ **RSET** <cadena1>=<cadena2>
Escribe los datos de la cadena2 en el registro de un fichero aleatorio indicado por cadena1, justificándolo por la derecha.

RUN [numlin]
Comienza la ejecución del programa a partir de la línea indicada. Si no se indica ninguna línea, se inicia al principio.

RUN "[dispositivo:][nombre[.ext]]"
Lee del dispositivo indicado un programa en ASCII y lo ejecuta posteriormente.

SAVE "[dispositivo:][nombre][.ext]]"[,A]
Graba el programa en memoria en el dispositivo indicado. Si se trata del cassette lo graba en formato ASCII. En el caso de disco (o RAM-DISK) lo graba en formato comprimido y en este caso se utiliza la opción A para indicar grabación en formato ASCII.

SCREEN [modo],[sprites],[click],[bauds],[impres],[entrelazado]
Selecciona el modo de pantalla, el tamaño de los sprites, el chasquido o no de las teclas al pulsarlas, la velocidad de grabación de la cinta, el tipo de impresora y, en los MSX-2, si se desea entrelazado para obtener así una mayor resolución.
(coor.X,coor.Y)

⊙ **SET ADJUST** (coor.X,coor.Y)
Ajusta la posición de la imagen en la pantalla, para adecuarla al tipo de monitor.
[sonido],[volumen]

⊙ **SET BEEP** [sonido],[volumen]
Selecciona el tipo del pitido BEEP y su volumen.

⊙ **SET DATE** cadena[,A]
Introduce la fecha en el reloj interno de los MSX-2. La opción A permite programar la alarma.

⊙ **SET PAGE** [visualización][,activa]
Selecciona las páginas de trabajo (activa) y de visualización (la que aparece en pantalla) entre las que existan en cada modo.

⊙ **SET PASSWORD** "clave"
Selecciona una contraseña para el sistema, de modo que sólo puedan utilizar el aparato aquellos que la conozcan previamente.

⊙ **SET PROMPT** "mensaje"
Este comando permite modificar el PROMPT del intérprete BASIC, que normalmente es un mensaje Ok.

⊙ **SET SCREEN** Almacena en la memoria permanente de los MSX-2 los valores actuales seleccionados con SCREEN, y los instaura nada más conectarse al aparato.

⊙ **SET TIME** <cadena>[,A]
Introduce la hora actual en el reloj interno de los MSX-2. la opción A permite programar la alarma.

⊙ **SET TITLE** ["título"],[color]
Almacena en la memoria permanente de los MSX-2 el título que aparecerá al conectar el aparato.

⊙ **SET VIDEO** [modo],[brillo],[color],[sinc],[sonido],[vídeo],[AV]
Programa los parámetros de trabajo para la superimposición y digitalización de vídeo, o la mezcla de sonido.

SGN (<num>)
Retorna en uno (1) si el valor es positivo, cero (Ø) si es cero, y menos uno (−1) si el valor es negativo.

SIN (<num>)
Retorna el valor del seno del número indicado, que debe estar expresado en radianes.

SOUND registro,<num>
Envía al registro del PSG indicado un cierto valor numérico.

SPACE$(<num>)
Retorna una cadena compuesta por tantos espacios en blanco como se indiquen.

SPC(<num>)
Retorna el número indicado de espacios en blanco. No se puede utilizar en formato cadena.

SPRITE ON|OFF|STOP
Conecta, desconecta o retiene las intrerrupciones generadas por colisiones entre sprites.

SPRITE$(<num>)=<cadena>
Define el sprite cuyo número se indica con los datos que componen la cadena especificada.

SQR(<num>)
Retorna la raíz cuadrada del valor dado.

STICK(<num>)
Retorna un número que indica el estado del joystick cuyo número se indica (Ø se refiere a las teclas de cursor del teclado).

STOP
Interrumpe la ejecución de un programa.

STOP ON|OFF|STOP
Conecta, desconecta o retiene las interrupciones generadas por la pulsación de CTRL+STOP.

STRIG(<num>)
Retorna un valor lógico TRUE (−1) si se está pulsando el disparo del joystick indicado o FALSE (Ø) en caso contrario.

STRIG ON|OFF|STOP
Conecta, desconecta o retiene las interrupciones generadas por la pulsación de los botones de disparo de los joystick.

STR$(<num>)
Retorna la cadena que representa el valor numérico indicado.

STRING$(<num>,<car>)
Retorna la cadena formada por la repetición del carácter indicado tantas veces como se especifique.

SWAP var1,var2
Intercambia los valores de las dos variables, que deben ser del mismo tipo.

TAB(<num>)
Mueve el cursor hacia la derecha hasta la posición horizontal indicada.

TAN(<num>)
Retorna el valor de la tangente del dato indicado, que debe estar expresado en radianes.

TIME
Variable del sistema que contiene el número de 1/5Ø de segundo transcurridos desde que se conectó la máquina. Puede modificarse su valor a voluntad.

TROFF
Desactiva el modo de tazado de ejecución, en que aparecen por pantalla los números de línea ejecutados.

TRON
Activa el modo de trazado de ejecución, en que aparecen por pantalla los números de línea ejecutados.

USR[<num1>](<nym2>)
Cede el control del programa a la rutina en código máquina definida por DEFUSR. Se puede indicar con num1 el número de rutina (Ø a 9) y con num2 pasarla un parámetro para su funcionamiento. Retorna un valor procedente de la rutina en código máquina.

VAL(<cadena>)
Retorna el valor explícito de una cadena de caracteres. Es la instrucción inversa a STR$.

VARPTR(<var>)
Retorna la dirección de memoria en que se almacena la variable indicada.

VARPTR(#numfich)
Retorna la dirección de inicio del FCB (bloque de control de fichero) asignado al fichero indicado.

VDP(<num>)
Variable del sistema que representa los contenidos de los diferentes registros del VDP. Modificando sus valores se modifican los de dichos registros.

VPEEK(dir)
Retorna el byte contenido en la dirección de memoria de vídeo indicada.

VPOKE dir,dato
Introduce el dato especificando en la dirección de la memoria de vídeo que se indica.

WAIT puerto,<num1>[?<num2>]
Espera hasta que la entrada del puerto indicado alcanza un valor, obtenido mediante la realización de ciertas operaciones lógicas con los operandos especificados.

WIDTH <num>
Especifica el número de caracteres que debe contener cada línea horizontal en modo texto.